

Notice of Allowability

Application No.

10/614,396

Examiner

MENGYAO ZHE

Applicant(s)

YAN ET AL.

Art Unit

2195

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included herewith (or previously mailed), a Notice of Allowance (PTOL-85) or other appropriate communication will be mailed in due course. **THIS NOTICE OF ALLOWABILITY IS NOT A GRANT OF PATENT RIGHTS.** This application is subject to withdrawal from issue at the initiative of the Office or upon petition by the applicant. See 37 CFR 1.313 and MPEP 1308.

1. ☒ This communication is responsive to 4/24/2008.
2. ☒ The allowed claim(s) is/are 1-26, 28-34, now renumbered as 1-33.
3. ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
 - a) ☐ All b) ☐ Some* c) ☐ None of the:
 1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this national stage application from the International Bureau (PCT Rule 17.2(a)).
 - * Certified copies not received: _____.

Applicant has THREE MONTHS FROM THE "MAILING DATE" of this communication to file a reply complying with the requirements noted below. Failure to timely comply will result in ABANDONMENT of this application.

THIS THREE-MONTH PERIOD IS NOT EXTENDABLE.

4. ☐ A SUBSTITUTE OATH OR DECLARATION must be submitted. Note the attached EXAMINER'S AMENDMENT or NOTICE OF INFORMAL PATENT APPLICATION (PTO-152) which gives reason(s) why the oath or declaration is deficient.
5. ☐ CORRECTED DRAWINGS (as "replacement sheets") must be submitted.
 - (a) ☐ including changes required by the Notice of Draftsperson's Patent Drawing Review (PTO-948) attached
 - 1) ☐ hereto or 2) ☐ to Paper No./Mail Date _____.
 - (b) ☐ including changes required by the attached Examiner's Amendment / Comment or in the Office action of Paper No./Mail Date _____.

Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the drawings in the front (not the back) of each sheet. Replacement sheet(s) should be labeled as such in the header according to 37 CFR 1.121(d).
6. ☐ DEPOSIT OF and/or INFORMATION about the deposit of BIOLOGICAL MATERIAL must be submitted. Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

Attachment(s)

1. ☐ Notice of References Cited (PTO-892)
2. ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3. ☐ Information Disclosure Statements (PTO/SB/08),
Paper No./Mail Date _____
4. ☐ Examiner's Comment Regarding Requirement for Deposit
of Biological Material
5. ☐ Notice of Informal Patent Application
6. ☒ Interview Summary (PTO-413),
Paper No./Mail Date _____.
7. ☒ Examiner's Amendment/Comment
8. ☐ Examiner's Statement of Reasons for Allowance
9. ☐ Other _____.

/Meng-Ai An/
Supervisory Patent Examiner, Art Unit 2195

EXAMINER'S AMENDMENT

1. An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to the applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it **MUST** be submitted no later than the payment of the issue fee.
2. An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it **MUST** be submitted no later than the payment of the issue fee.
3. Authorization for this examiner's amendment was given in a telephone interview with Mr. Chet J. Bonner on 5/19/2008.
4. Please replace all of the claims with the following of applicant's amended claims.
 1. (Currently Amended) A method for maintaining an external module type definition table by a statically configured portion of an operating system kernel that is executable by a processor, comprising:
~~instructions in the static operating system kernel for:~~
identifying a module,
identifying a module type of the module;

searching an external module type definition table that is external to the statically configured portion of the operating system kernel for the module type;

determining the module type is not defined in the external module type definition table;

dynamically creating a module type definition including at least one support module identifier; and

updating the external module type definition table to include the dynamically created module type definition and;

loading the module based upon the module type and at least one support module identified by the support module identifier without reloading the statically configured portion of the operating kernel.

2. (Currently Amended) The method of claim 1, wherein dynamically creating a module type definition includes receiving an operator generated dynamically loadable kernel module (“DLKM”) type identifier.

3. (Currently Amended) The method of claim 1, wherein dynamically creating a module type definition includes receiving a computer generated dynamically loadable kernel module (“DLKM”) type identifier.

4. (Original) The method of claim 1, wherein creating a module type definition includes receiving at least one support module identifier associated with control logic operative to conduct pre-registration support.

5. (Original) The method of claim 1, wherein creating a module type definition includes receiving at least one support module identifier associated with control logic operative to conduct a registration function.

6. (Original) The method of claim 1, wherein creating a module type definition includes receiving at least one support module identifier associated with control logic operative to conduct post-registration support.

7. (Original) The method of claim 1, wherein creating a module type definition includes receiving at least one support module identifier associated with control logic operative to conduct pre-loading support.

8. (Original) The method of claim 1, wherein creating a module type definition includes receiving at least one support module identifier associated with control logic operative to conduct post-loading support.

9. (Original) The method of claim 1, wherein creating a module type definition includes receiving at least one of a pointer and a reference, each at least one of a pointer and a reference being respectively associated with a support module.

10. (Original) The method of claim 1, wherein creating a module type definition includes receiving at least one symbol name, each symbol name being respectively associated with a support module.

11. (Currently Amended) A system including a processor for maintaining an external module type definition table that is external to a statically configured portion of an operating system kernel, comprising:

module type detection logic on a computer readable medium for detecting that a module is of an undefined module type;

module type identification logic on the computer readable medium for assigning a new module type associated with the module;

support module identification logic on the computer readable medium for identifying at least one support module ~~associated with~~ used to support the module type;

support module loading logic on the computer readable medium for loading the at least one identified support module; ~~and~~

module type definition logic on the computer readable medium for dynamically defining the module type as a function of the module type and for externally storing the

Art Unit: 2195

module type data, including data identifying at least one support module ~~associated with~~
used to support the module type, thereby updating the external module type definition
table; and

logic for loading the module without reloading the statically configured portion of
the operating kernel.

12. (Currently Amended) The system of claim 11, wherein the module type identification logic includes logic for receiving an operator generated dynamically loadable kernel module (~~“DLKM”~~) type identifier.

13. (Currently Amended) The system of claim 11, wherein the module type identification logic includes logic for receiving a computer generated dynamically loadable kernel module (~~“DLKM”~~) type identifier.

14. (Original) The system of claim 11, wherein the support module identification logic includes logic for identifying a support module operative to conduct pre-registration support.

15. (Original) The system of claim 11, wherein the support module identification logic includes logic for identifying a support module operative to conduct a registration function.

16. (Original) The system of claim 11, wherein the support module identification logic includes logic for identifying a support module operative to conduct post-registration support.

17. (Original) The system of claim 11, wherein the support module identification logic includes logic for identifying a support module operative to conduct pre-loading support.

18. (Original) The system of claim 11, wherein the support module identification logic includes logic for identifying a support module operative to conduct post-loading support.

19. (Original) The system of claim 11, wherein the support module identification logic is operative to receive at least one of a pointer and a reference, each at least one of a pointer and a reference being respectively associated with each of the at least one support module.

20. (Original) The system of claim 11, wherein the support module identification logic is operative to receive at least one symbol name, each symbol name being respectively associated with each of the at least one support module.

21. (Currently Amended) A computer-readable storage medium encoded with processing instructions executable by a processor for maintaining an external module type definition table for use by a statically configured portion of an operating system kernel, comprising:

- instructions to identify a module type of a first module;
- instructions to determine that the module type of the first module is undefined;
- instructions to identify data defining the module type of the first module; and
- instructions to identify a support module ~~associated with~~ used to support the first module type;

- instructions to store the data defining the module type, including the ~~associated~~ support module that is used to support the module type, in the external module type definition table ~~in a location~~ that is external to the statically configured portion of the operating system kernel; and

- instructions to load the first module based upon the module type without reloading the statically configured portion of the operating kernel.

22. (Currently Amended) A The computer-readable storage medium of claim 21 wherein the data defining the module type comprises a pre-loading support module.

23. (Currently Amended) A The computer-readable storage medium of claim 21 wherein the data defining the module type comprises a post-loading support module.

24. (Currently Amended) A The computer-readable storage medium of claim 21 wherein the data defining the module type comprises a pre-registration support module.

25. (Currently Amended) A The computer-readable storage medium of claim 21 wherein the data defining the module type comprises a post-registration support module.

26. (Currently Amended) A n static operating kernel on a computer readable medium, the operating kernel being executable by a processor, comprising:

- logic on the computer readable medium to receive a request to load a module,
- logic on the computer readable medium to identify a module type of the module,
- logic on the computer readable medium to reference an external module type definition table that is external to a statically configured portion of the operating system kernel,

- logic on the computer readable medium to identify at least one support module ~~associated with~~ used to support the module type in the external module type definition table,

- logic on the computer readable medium to load the module based upon the module type and the at least one support module associated with the module type without reloading the statically configured portion of the operating kernel,

- logic on a computer readable medium for identifying at least one module type not previously defined in the external module type definition table,

- logic on the computer readable medium to dynamically define the at least one module type, the definition including an identifier for at least on support module,

- logic on the computer readable medium to update the external module type definition table with the dynamically defined at least one module type.

27. (Canceled).

28. (Currently Amended) The ~~static~~ operating kernel of claim 26, where the logic to dynamically define the at least one external module type includes receiving an operator identified module type.

29. (Currently Amended) The ~~static~~ operating kernel of claim 26, where the logic to dynamically define the at least one external module type includes receiving at least one identified support modules from an operator.

30. (Currently Amended) An static operating system kernel on a computer readable medium, the operating system kernel being executable by a processor, comprising:

- logic on the computer readable medium to receive a request to load a module,
- logic on the computer readable medium to identify a module type associated with the module,

- logic on the computer readable medium to reference an external module type reference table that is external to the statically configured portion of the operating kernel,

- logic on the computer readable medium to determine the module type is undefined in the external module type reference table,

- means on the computer readable medium to identify at least one support module used to support the module type and associate that support module with a definition of the module type in the external module type reference table, and

- means on the computer readable medium to dynamically load the module without reloading the statically configured portion of the operating system kernel.

31. (Currently Amended) The ~~static~~ operating system kernel of claim 30, wherein the means to dynamically load the module comprises:

- means to dynamically define the module type that is undefined in the module type reference table,

- means to dynamically update the external module type definition table to include the defined module type,

logic to load the module based upon the updated external module type definition table.

32. (Currently Amended) The ~~static~~ operating system kernel of claim 31, wherein the means to dynamically define the module type that is undefined in the module type reference table comprises logic on the computer readable medium to receive at least one operator generated module type.

33. (Currently Amended) The ~~static~~ operating system kernel of claim 31, wherein the means to dynamically define the module type that is undefined in the module type reference table comprises logic on the computer readable medium to receive at least one software generated module type.

34. (Currently Amended) The ~~static~~ operating system kernel of claim 31, wherein the means to dynamically define the module type that is undefined in the module type reference table comprises logic on the computer readable medium to identify at least one support module associated with the module type.

5. Any inquiry concerning this communication or earlier communications from the examiner should be directed to MengYao Zhe whose telephone number is 571-272-6946. The examiner can normally be reached on Monday Through Friday, 10:00 - 8:00 EST. If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng-Ai An can be reached at 571-272-3756. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

/Meng-Ai An/

Supervisory Patent Examiner, Art Unit 2195